

Федеральное государственное бюджетное образовательное учреждение высшего образования



**«Московский государственный технический
университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)**

ФАКУЛЬТЕТ ФУНДАМЕНТАЛЬНЫЕ НАУКИ

КАФЕДРА МАТЕМАТИКА И КОМПЬЮТЕРНЫЕ НАУКИ

Отчет

по лабораторному заданию № 14

Вариант 18

Дисциплина: Информатика

Название лабораторного задания: Модульное программирование.

Использование подпрограмм с параметрами сложных структурных типов данных в C++.

Студент гр. ФН11-22Б

80 17.03.21.
(Подпись, дата)

М.Х. Хаписов
(И.О. Фамилия)

Преподаватель Доцент кафедры ИУ-6

(Подпись, дата)

Т.Н. Ничушкина
(И.О. Фамилия)

17.3.21

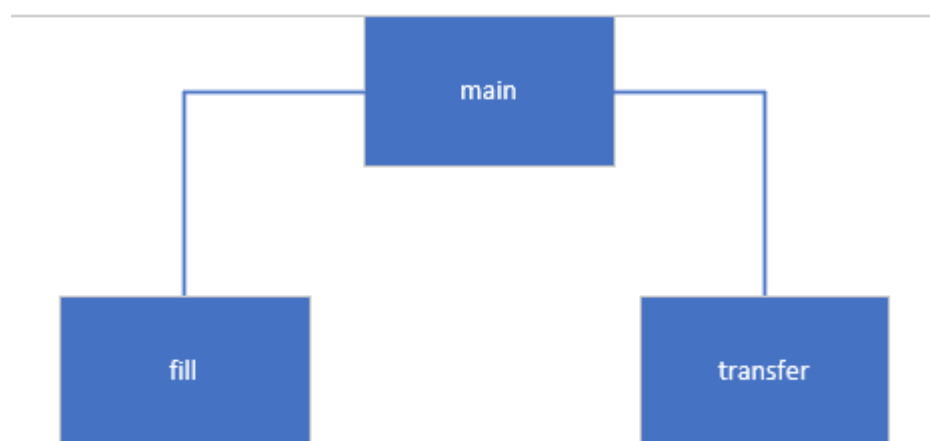
Москва, 2021

Цель работы: научиться использовать подпрограммы с параметрами сложных структурных типов данных в C++

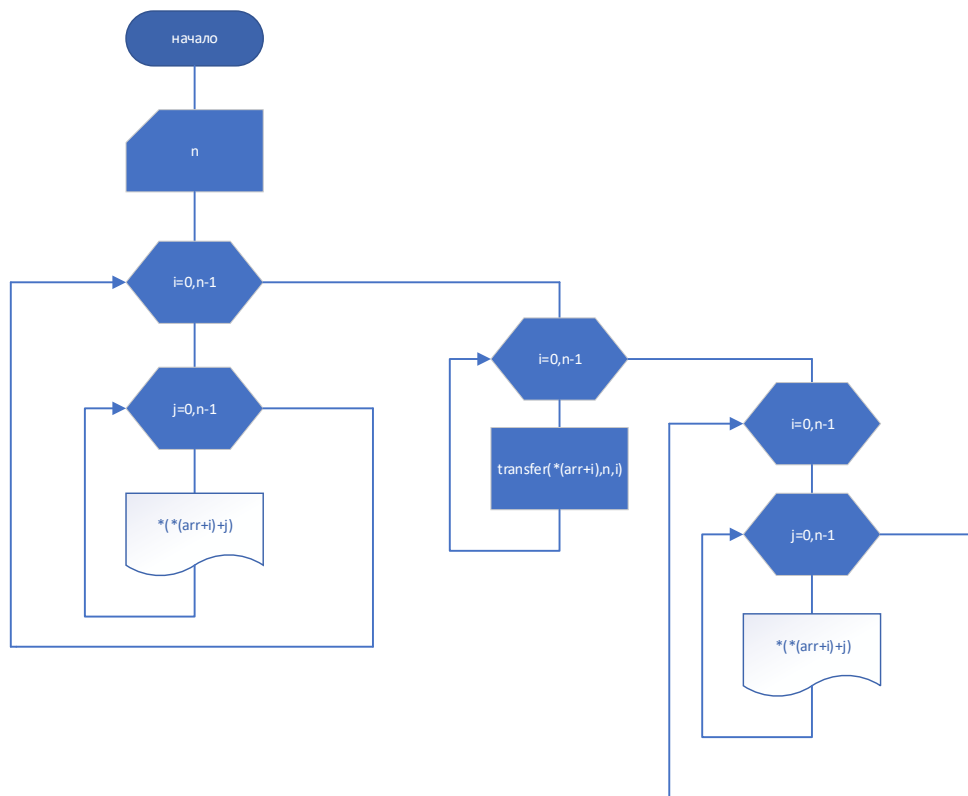
Задание: Разработать модуль, содержащий указанные процедуры и функции. Написать тестирующую программу.

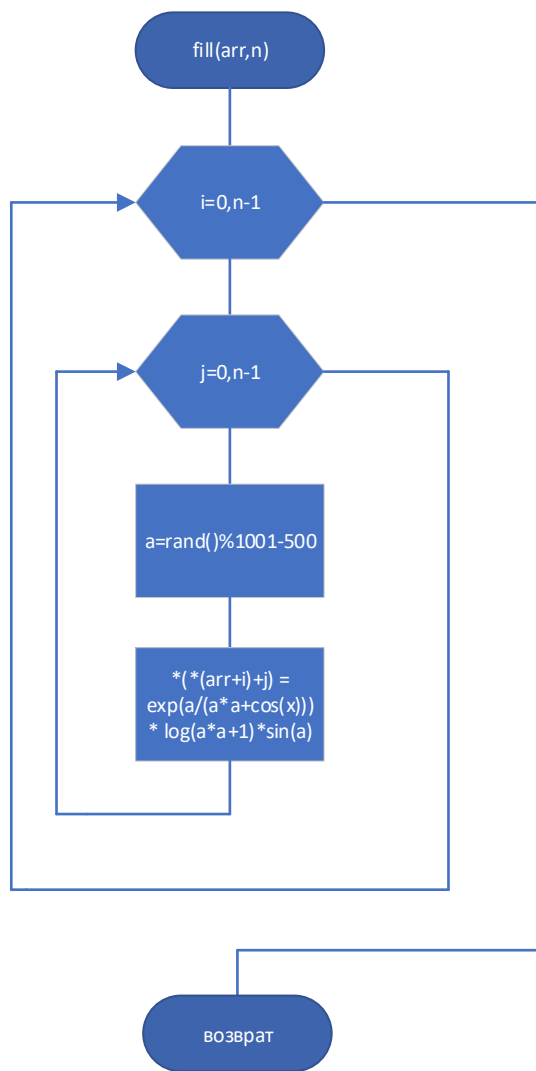
Составить подпрограмму, которая для i -ой строки матрицы ($N \times N$) вычисляет среднее арифметическое значение отрицательных элементов и помещает его на место последнего элемента этой же строки. Если отрицательных элементов нет, на место этого элемента помещается 0. Применить подпрограмму для преобразования всех строк матрицы, содержащей случайные вещественные числа. Вывести на экран исходную и преобразованную матрицу и всю найденную информацию с соответствующими комментариями.

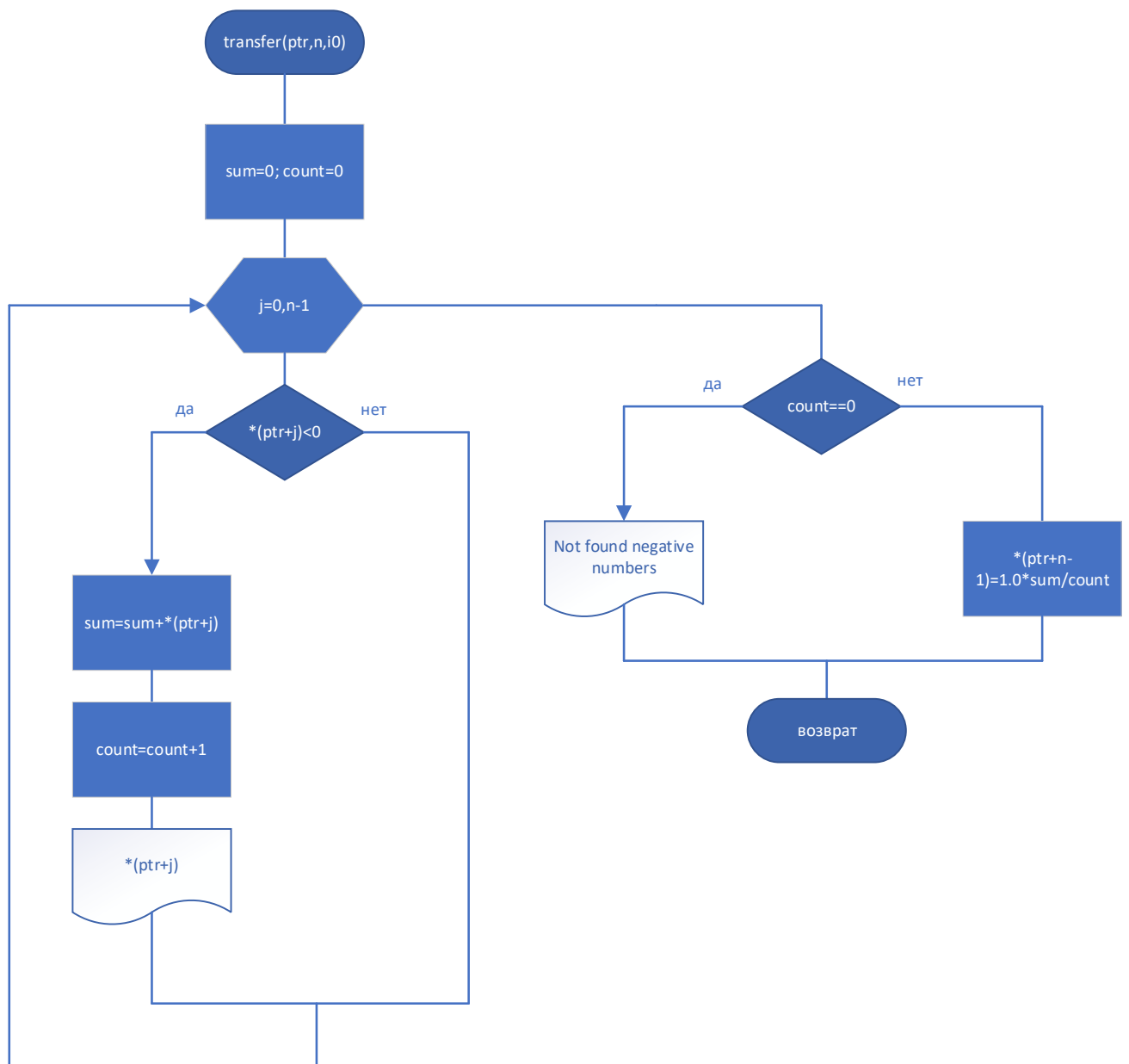
Структурная схема



Схемы алгоритмов







Текст программы

Файл *main.cpp*

```

#include "Header.h"
void main() {
    int n;
    puts("Input size of a matrix");
    std::cin>>n;
    while(n<=0) {
        puts("Error! Size of the matrix must be positive");
        std::cin>>n;
    }
    double **arr=new double*[n];
    for(size_t i=0;i<n;i++) *(arr+i)=new double[n];
    fill(arr,n);
    puts("The initial matrix");
    for(size_t i=0;i<n;i++) {
        for(size_t j=0;j<n;j++) printf("%9.5lf ",*(*(arr+i)+j));
    }
}

```

```

        puts("");
    }
    puts("");
    for(size_t i=0;i<n;i++) transfer(*(arr+i),n,i);
    puts("");
    puts("The transformed matrix");
    for(size_t i=0;i<n;i++) {
        for(size_t j=0;j<n;j++) printf("%9.51f ",*(*(arr+i)+j));
        puts("");
    }
    system("pause");
}

```

Файл Header.h

```

#pragma once
#include <iostream>
#include <ctime>
#include <cmath>
void fill(double **arr,int n);
void transfer(double *ptr,int n,int i0);

```

Файл Source.cpp

```

#include "Header.h"
void fill(double **arr,int n) {
    srand(time(nullptr));
    for(size_t i=0;i<n;i++) {
        for(size_t j=0;j<n;j++) {
            double a=rand()%1001-500;
            *(*(arr+i)+j)=exp(a/(a*a*cos(a)))*log(a*a+1)*sin(a);
        }
    }
}
void transfer(double *ptr,int n,int i0) {
    double sum=0;
    size_t count=0;
    for(size_t j=0;j<n;j++) {
        if(*(ptr+j)<0) {
            if(count==0) printf("In the %d row:",i0+1);
            sum+=*(ptr+j);
            count++;
            printf(" %g",*(ptr+j));
        }
    }
    if(count==0) printf("In %d row not found negative elements, so the last column isn't changed\n",i0+1);
    else {
        printf(" = %g; average = %g\n",sum,1.0*sum/count);
        *(ptr+n-1)=1.0*sum/count;
    }
}

```

Тестирование

C:\Users\Knigan\source\repos\Project2\Debug\Project2.exe

Input size of a matrix

5

The initial matrix

-3.56810	-4.54155	-11.05958	-2.82199	5.72843
-9.56819	4.37637	1.53074	-9.91518	1.87151
10.79106	3.15798	3.21731	1.39175	-3.08156
-10.95362	-8.16006	-5.38215	2.95919	0.00035
-10.58419	-8.61376	9.14608	-3.59947	-4.98608

Address of the -3.56810 is 01104AE0

Address of the -4.54155 is 01104AE8

Address of the -11.05958 is 01104AF0

Address of the -2.82199 is 01104AF8

Sum in the 1 row is -21.99121, average = -5.49780

Address of the -9.56819 is 01105668

Address of the -9.91518 is 01105680

Sum in the 2 row is -19.48337, average = -9.74168

Address of the -3.08156 is 0110CF38

Sum in the 3 row is -3.08156, average = -3.08156

Address of the -10.95362 is 0110CF70

Address of the -8.16006 is 0110CF78

Address of the -5.38215 is 0110CF80

Sum in the 4 row is -24.49583, average = -8.16528

Address of the -10.58419 is 011100F8

Address of the -8.61376 is 01110100

Address of the -3.59947 is 01110110

Address of the -4.98608 is 01110118

Sum in the 5 row is -27.78350, average = -6.94587

The transformed matrix

-3.56810	-4.54155	-11.05958	-2.82199	-5.49780
-9.56819	4.37637	1.53074	-9.91518	-9.74168
10.79106	3.15798	3.21731	1.39175	-3.08156
-10.95362	-8.16006	-5.38215	2.95919	-8.16528
-10.58419	-8.61376	9.14608	-3.59947	-6.94587

Для продолжения нажмите любую клавишу . . . ■

В 4-ой строке отрицательные числа расположены рядом друг с другом (соседние) и их адреса отличаются на 8. Соседние же элементы в столбце (4 и 5 строки) расположены на расстоянии в 12680, что говорит о том, что элементы 5-ой строки располагаются в памяти отдельно от элементов 4-ой строки.

```
Консоль отладки Microsoft Visual Studio
Input size of a matrix
6
The initial matrix
-3.28173  10.15835  9.79391  -9.77605  -11.80663  -6.33209
 5.75243  6.29299 -10.92768  9.53502  -8.76783  4.97383
-4.83933  -0.30160  -0.72964  10.25138  4.68793  10.89684
-7.76315  4.49868  7.67377  10.00761  -9.50144  11.67892
 7.09426  -2.30254  8.33017  -2.77713  11.08366  -7.66345
-11.40182 10.24543  7.14223  5.62080 -10.88490  -9.46499

Address of the -3.28173 is 007C4AE0
Address of the -9.77605 is 007C4AF8
Address of the -11.80663 is 007C4B00
Address of the -6.33209 is 007C4B08
Sum in the 1 row is -31.19650, average = -7.79913

Address of the -10.92768 is 007C5678
Address of the -8.76783 is 007C5688
Sum in the 2 row is -19.69552, average = -9.84776

Address of the -4.83933 is 007CCF20
Address of the -0.30160 is 007CCF28
Address of the -0.72964 is 007CCF30
Sum in the 3 row is -5.87058, average = -1.95686

Address of the -7.76315 is 007CCF80
Address of the -9.50144 is 007CCFA0
Sum in the 4 row is -17.26459, average = -8.63229

Address of the -2.30254 is 007D0910
Address of the -2.77713 is 007D0920
Address of the -7.66345 is 007D0930
Sum in the 5 row is -12.74312, average = -4.24771

Address of the -11.40182 is 007D0968
Address of the -10.88490 is 007D0988
Address of the -9.46499 is 007D0990
Sum in the 6 row is -31.75171, average = -10.58390

The transformed matrix
-3.28173  10.15835  9.79391  -9.77605  -11.80663  -7.79913
 5.75243  6.29299 -10.92768  9.53502  -8.76783  -9.84776
-4.83933  -0.30160  -0.72964  10.25138  4.68793  -1.95686
-7.76315  4.49868  7.67377  10.00761  -9.50144  -8.63229
 7.09426  -2.30254  8.33017  -2.77713  11.08366  -4.24771
-11.40182 10.24543  7.14223  5.62080 -10.88490 -10.58390
Для продолжения нажмите любую клавишу . . .
```


C:\Users\Knigan\source\repos\Project2\Debug\Project2.exe

Input size of a matrix

4

The initial matrix

0.43080	-9.93357	-7.56940	10.64694
-10.43610	5.12265	11.49086	8.71595
-9.93357	-10.48859	-11.73342	-11.87539
-0.94926	7.46721	-11.26688	10.88226

Address of the -9.93357 is 00A54AE8

Address of the -7.56940 is 00A54AF0

Sum in the 1 row is -17.50297, average = -8.75149

Address of the -10.43610 is 00A60628

Sum in the 2 row is -10.43610, average = -10.43610

Address of the -9.93357 is 00A60448

Address of the -10.48859 is 00A60450

Address of the -11.73342 is 00A60458

Address of the -11.87539 is 00A60460

Sum in the 3 row is -44.03097, average = -11.00774

Address of the -0.94926 is 00A60358

Address of the -11.26688 is 00A60368

Sum in the 4 row is -12.21614, average = -6.10807

The transformed matrix

0.43080	-9.93357	-7.56940	-8.75149
-10.43610	5.12265	11.49086	-10.43610
-9.93357	-10.48859	-11.73342	-11.00774
-0.94926	7.46721	-11.26688	-6.10807

Для продолжения нажмите любую клавишу . . . ■

C:\Users\Knigan\source\repos\Project2\Debug\Project2.exe

Input size of a matrix

3

The initial matrix

1.65474	6.10529	-4.65154
-2.38757	6.36657	9.13657
-4.85905	4.21386	-6.42047

Address of the -4.65154 is 00C80718

Sum in the 1 row is -4.65154, average = -4.65154

Address of the -2.38757 is 00C80828

Sum in the 2 row is -2.38757, average = -2.38757

Address of the -4.85905 is 00C801F8

Address of the -6.42047 is 00C80208

Sum in the 3 row is -11.27952, average = -5.63976

The transformed matrix

1.65474	6.10529	-4.65154
-2.38757	6.36657	-2.38757
-4.85905	4.21386	-5.63976

Для продолжения нажмите любую клавишу . . .

```

C:\Users\Knigan\source\repos\Project2\Debug\Project2.exe
Input size of a matrix
-2
Error! Size of the matrix must be positive
-1
Error! Size of the matrix must be positive
0
Error! Size of the matrix must be positive
2
The initial matrix
-8.51911  3.71279
 0.57632  0.10216

Address of the -8.51911 is 009249B8
Sum in the 1 row is -8.51911, average = -8.51911

In 2 row not found negative elements, so the last column isn't changed

The transformed matrix
-8.51911 -8.51911
 0.57632  0.10216
Для продолжения нажмите любую клавишу . . .

```

Вывод: я научился использовать подпрограммы с параметрами сложных структурных типов данных в C++

Ответы на контрольные вопросы:

1. Что такое модуль в C++ и какова его структура? (Ответ: Среда Visual C++ позволяет создавать и отлаживать программы, использующие не только стандартные, но и пользовательские библиотеки подпрограмм – модули. Модуль C++ обычно включает два файла: заголовочный файл с расширением «.h» и файл реализации с расширением «.cpp»)
2. Что такое заголовочный файл и как его подключить к программе? (Ответ: Заголовочный файл играет роль интерфейсной секции модуля. В него помещают объявление экспортируемых ресурсов модуля: • прототипы (заголовки) процедур и функций, • объявление переменных, типов и констант. Заголовочный файл подключают командой `#include ".h"`, записываемой в

файле реализации программы или другого модуля, если они используют ресурсы описываемого модуля)

3. Как передать в подпрограмму многомерный массив произвольного размера? (Ответ: с помощью передачи в подпрограмму параметров-указателей)
4. Что такое объявление и описание функции и чем они отличаются? Что такое прототип функции? (Ответ: объявление функции – это описание её имени, типа возвращаемого ей значения и типов принимаемых ею параметров. Описание функции – это описание тела функции. Прототип функции – это объявление функции без её описания (описание, как правило, происходит далее в тексте программы))
5. Назовите способы передачи данных в функции. (Ответ: можно передавать параметры в функцию по значению, по ссылке и по указателю)